

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET AUTOMATIQUE

Interactive Axial Deformations

Francis Lazarus, Sabine Coquillart and Pierre Jancène

N° 1891

Avril 1993

PROGRAMME 4

Robotique,

image

et vision



***rapport
de recherche***

1994



Interactive Axial Deformations

Francis Lazarus, Sabine Coquillart and Pierre Jancène*

Programme 4 — Robotique, image et vision

Projet Syntim

Rapport de recherche n° 1891 — Avril 1993 — 19 pages

Abstract: This paper presents an interactive deformation technique. The entity employed for defining the deformation of an object is a 3D axis as well as some associated parameters.

The technique allows an easy specification and control of deformations that can be defined with that entity such as bending, twisting and scaling.

Contrary to existing techniques, the method developed is independent of both the geometric model of the object to be deformed and the creation technique used to define the object.

Moreover, it can easily be integrated into traditional interactive modeling systems.

Key-words: CR Categories and Subject Descriptors: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling - Curve, surface, solid, and object representation; Geometric algorithms, languages, and systems; Hierarchy and geometric transformations; I.3.6 [Computer Graphics]: Methodology and Techniques - Interaction techniques.

(Résumé : tsvp)

*Email : {Francis.Lazarus,Sabine.Coquillart,Pierre.Jancene}@inria.fr

Unité de recherche INRIA Rocquencourt

Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)

Téléphone : (33 1) 39 63 55 11 – Télécopie : (33 1) 39 63 53 30

Déformation Axiales Interactives

Résumé : Cet article présente une technique interactive de déformation. L'entité employée pour définir la déformation d'un objet est un axe tridimensionnel auquel sont associés certains paramètres.

Notre technique permet de spécifier et contrôler simplement les déformations qui peuvent être définies avec cette entité, telles que la courbure, la torsion et la dilatation.

Contrairement aux techniques existantes, la méthode développée est indépendante aussi bien du modèle géométrique de l'objet à déformer que de la technique de création utilisée pour définir l'objet.

De plus, elle peut aisément être intégrée dans un modèleur traditionnel.

1 Introduction

Today's graphics systems for producing images consist of several software components, the first one being a 3D geometric modeler with which a user can define objects geometry.

A geometric modeler often includes two classes of modeling techniques: the creation techniques such as the sweep, the loft or the extrusion and the modification techniques. Modification, or deformation, techniques may be used either for refining the shape of an object defined using creation techniques or for changing the shape of existing objects obtained by scanning a real object, for instance.

The approaches taken to deforming objects include:

- Manipulation of the geometric model.

The deformation techniques of this class consist of direct manipulation of the geometric model representing the surface. They are often very dependent on the geometric model. For instance, one may interactively move the control points of a spline surface or of a hierarchical spline surface [FB88]. If the geometric model representing the surface is an implicit surface [WBB⁺90] defined by points (resp. axis), the deformation technique involves moving the points (resp. deforming the axis).

Recent work has also shown that the trivial solution that consists of moving the control points of a spline surface can be extended to allow the user to manipulate freely any point or even a curve of the surface [BB91, WW92]. These new techniques make direct manipulation less dependent on the geometric model.

- Manipulation of a creation entity.

In a general modeling system, it is usually preferable to use the same geometric model such as a spline or a polygonal surface, for representing each object. In this case the creation technique and thus the creation entities used for defining an object are often independent of the geometric model of the object. Creation entities may include the axis or the cross-section used for creating a surface as a sweep, or the profile curve, or

angle used for the defining of a surface as a surface of revolution. A common practice for deforming objects this way consists of manipulating the entities, or the parameters, used to create the surface.

Basically, deformations of this class can be resumed as a redefinition of some of the creation entities followed by a recomputation of the surface.

- Manipulation of a deformation entity.

Techniques for which a deformation entity is defined and used to deform the object fall into this class. The deformation entity can be compared to the creation entity, since it plays the role of an interface between the geometric model and the user. One of its consequences is to make the deformation technique independent of both the geometric model and the creation technique. Techniques such as the FFD and the EFFD [SP86, Coq90], or Cobb's region or skeletal warp [Cob84] lie in that class.

Techniques of the third class present several advantages:

- From a user point of view, the same deformation technique can be applied to any object, no matter where it comes from. Furthermore, the geometric model becomes transparent for the user.
- These techniques can be combined with one another to increase the power of the modeling system.
- The deformation is completely defined by the deformation entity. A deformation tool [Coq90] can thus be defined, permitting the reproduction of the deformation several times, on possibly different surfaces.

In order to take advantage of these benefits, one of our goals was extending some deformation techniques of the first two classes to the third one.

This paper deals with deformations defined by an axis. These deformations are simple and specific, but very useful and commonly employed. They include bending, twisting and scaling around an axis.

Several techniques exist for deforming an object with an axis.

One of them consists of defining the object as a sweep surface and then deforming the axis or modifying other parameters such as a scale factor or a twist factor defined along the axis (cf. Manipulation of a creation entity). A second solution consists of using implicit surfaces, defined by a skeleton-axis, to represent the surface (cf. Manipulation of the geometric model).

It follows that:

- The object has to be defined as either a sweep surface or an implicit surface with an axis as skeleton. The class of objects that can be defined this way is rather restricted.
- The only axis that may be used to deform such an object is the creation axis.
- Combining these deformations with other deformations is rarely possible.

Parent [Par77] uses axial deformations for deforming 2D shapes while Barr [Bar84], has proposed 3D axial deformations where the axes considered are only straight axes and the bending operations are very restricted in comparison with what we can expect from a real 3D axis.

Our purpose is thus to develop an axial deformation technique that is independent of the geometric model and valid for any creation technique and any axis.

The following section explains the principles of the Axial Deformation technique and emphasizes the computation process. Section 3 presents several extensions increasing both the generality and the power of the previously defined deformation techniques. Finally, we give some examples to illustrate our approach.

2 Axial Deformations

Our goal is to define a deformation technique that makes use of a 3D axis for deforming existing objects. This technique is called AxDf, for Axial Deformation. Suppose we have an object. From the user's point of view, the deformation process is as follows:

- First, the user defines a 3D axis, that can be positioned either inside or outside the object. This 3D axis may have any shape, depending on the deformation desired.
- Second, the user changes the shape of the axis, and the deformations so applied to the axis are automatically passed on to the object.

Figure 1 illustrates the AxDf technique by a simple example. Figure 1-left shows the sphere we wish to deform as well as the axis used for defining the deformation; a straight axis, in black has been designed. Figure 1-right shows the sphere deformed in a manner that is intuitively consistent with the motion of the axis visualized in black.

Within an interactive modeling system, the 3D axis will be represented by a 3D curve; any spline curve can be used.

To make this deformation technique practical, we must find a way to pass the axis' deformations to the object. We handle this problem using a two-step process. First, each vertex ¹ V of the object is attached to one point A_V of the axis and its local coordinates (x, y, z) in the axis' local coordinate system are computed. Second, the deformed vertex is obtained by computing the associated local coordinate system at A'_V , homologous to A_V on the deformed axis, and transforming the (x, y, z) coordinates from this coordinate system to the world coordinate system.

Three problems have to be considered:

- attaching a vertex to the axis,

¹From now on, every point of an object will be called a vertex to avoid any confusion with the points of the deformation axis.

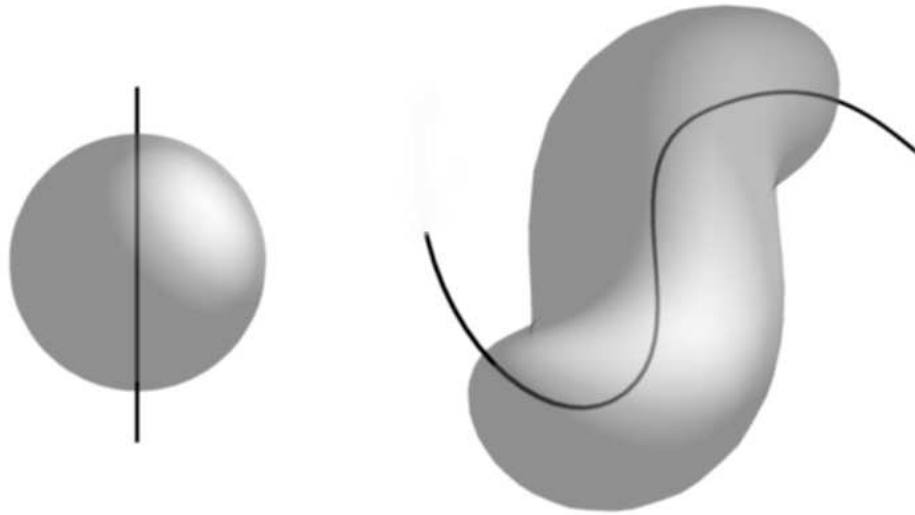


Figure 1: Deforming a sphere.

- defining local coordinate frames on the axis,
- computing the coordinates of the deformed vertices.

These three problems will be studied in the following paragraphs.

2.1 Attaching a vertex to the axis

To compute axial deformations, each object vertex must be attached to the 3D axis. A natural association consists of attaching the vertex to the closest point of the curve. The closest point can be computed either recursively by using the convex hull property of a spline curve or by discretizing the curve. Selecting the closest point of the curve may raise some problems, namely when several points of the axis are located at equal distance from the object vertex, or when a vertex lies in two different planes normal to the axis and the best

choice is not the closest axis point. Several techniques for improving that choice are under consideration. One of them consists of taking into account vertex adjacencies in order to make the association function continuous.

The attach point is represented by its parameter value on the axis curve.

2.2 Axis local coordinate frames

After attaching each object vertex to a point of the axis, we now define a local coordinate frame at each point of the axis in order to allow the computation of the (x, y, z) local coordinates of each object vertex in this coordinate system.

Several methods exist for defining coordinate frames at each point of a 3D curve. A common practice consists of considering the well known Frenet frame. For each point of a 3D curve, the Frenet frame is represented by the three orthogonal unit vectors defined by the tangent, the normal and the binormal at that point. This frame depends on the first and the second derivatives of the curve, it can thus be computed explicitly.

However, three problems exist:

- The normal is not defined on linear curve segments, nor more generally, where the curvature vanishes.
- The normal direction flips at the inflection points.
- The normal can rotate in an undesirable manner around the 3D curve.

An alternative to the Frenet frame has been proposed by Klok in [Klo86]. Klok defines a rotation minimizing orthogonal frame by the requirement that the rotation of the frame be minimized along the curve.

Assuming that $C(s)$, $s \in [0, L]$ is a regular curve, Klok defines the rotation minimizing orthogonal frame $t(s), f(s), g(s)$ along C such that

$$t(s) = C'(s) / \|C'(s)\|$$

$$f'(s) = -(C''(s).f(s))C'(s)/||C'(s)||^2$$

$$g'(s) = -(C''(s).g(s))C'(s)/||C'(s)||^2$$

$f(0)$ and $g(0)$ being chosen such that $t(0)$, $f(0)$ and $g(0)$ are mutually orthogonal unit vectors.

These equations guarantee that the rotation of $f(s)$ is minimal along the curve.

As a closed-form solution of the previous equations does not exist, Klok also gives a geometric construction of the rotation minimizing frame based on an approximation of the curve by a sequence of straight line segments.

This solution has been adopted for computing the frames at each point of the axis. Note that a special treatment has been added for cusps: when a cusp is detected on the axis, the orientation of the following frames is reversed.

The (x, y, z) coordinates of V are thus defined as the coordinates of V in the frame associated to A_V . Note that in most cases, x equals zero.

2.3 Deformed vertices

The process for computing the deformed position V_{AxDf} of a vertex V is as follows.

Let A'_V be the point of the deformed axis corresponding to A_V . A'_V is defined such that its parameter value on the deformed axis is equal to the previously computed parameter value of A_V .

V_{AxDf} is the vertex defined by the (x, y, z) local coordinates in the frame associated with A'_V .

3 Extensions

We have thus far proposed a basic version of the AxDf technique. The only deformations that can be controlled are the deformations obtained either by bending or stretching the axis (cf. Figure 1 for a simple example). Other parameters can be defined in order to extend the AxDf technique. Some of them are suggested in the following paragraphs.

3.1 Scale and twist

Scale and twist graphs can easily be added to the Axial Deformation technique. The twist (resp. scale) value permits the twisting (resp. scaling) of the object around the axis. The objective is to define both a twist and a scale factor at each point of the axis; these factors are then used for computing the deformed object's vertices. From the user's point of view, a twist (resp. scale) factor can be defined at any point of the axis. The value along the axis is thus obtained by interpolating the values defined by the user. Figure 2 illustrates the twist factor by a simple example. The undeformed object is composed of two rods shown in Figure 2-left; a straight axis has been positioned between the 2 rods. Figure 2-right is obtained without changing the shape of the axis, just by adding a twist factor of 0 at one end and another of 360 degrees at the other end.

3.2 Zone of influence

In order to improve the Axial Deformation technique, a zone of influence can be introduced to define the portion of the 3D space to be deformed. In our implementation, we have taken advantage of the deformation axis to define that space. A simple solution consists of defining two zones of influence ZI_{min} and ZI_{max} as general cylinders around the deformation axis, ZI_{min} being included into ZI_{max} .

The vertices of the object lying inside ZI_{min} (resp. outside ZI_{max}) will be fully deformed (resp. will not be deformed at all). Vertices in-between will be partially deformed by interpolating the deformation parameters, such as twist, scale and attach point.

Each zone of influence is defined by the two radii R_{min} and R_{max} of each circular cross-section along the axis. These radii are defined in the same manner as the twist or the scale parameters.

Figure 3 illustrates the use of the zone of influence. In Figure 3-left, a planar surface is shown and the axis used to deform the surface is visualized in black. The two zones of influence are also visualized using transparencies: ZI_{min} is the inner zone while ZI_{max} is



Figure 2: Twisting two straight rods.

the outer one. Only the vertices of the object lying inside Z_{Imax} will be deformed. The deformed surface is shown in Figure 3-right where the deformed axis is shown in black.

3.3 Deformation tool

A consequence of the independence of the AxDf technique and the geometric model is the capacity of exploiting the deformation tool paradigm. The deformation tool must fully describe the deformation. In AxDf, a deformation tool is composed of:

- two axes: the undeformed, or initial axis, and the deformed, or final one,
- two zones of influence, Z_{Imin} and Z_{Imax} ,
- a twist graph and a scale graph.

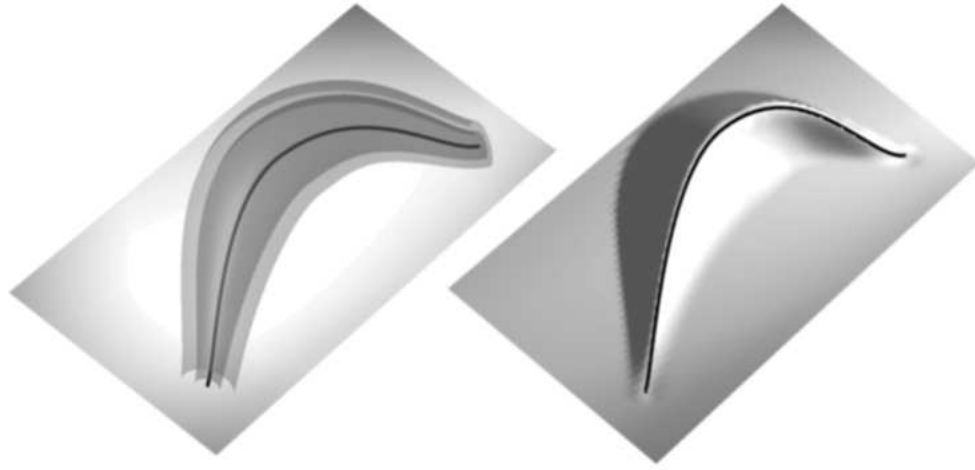


Figure 3: Zone of influence.

As the deformation is fully described by the deformation tool, the same deformation can be applied several times to different objects, simply by using the same deformation tool. In Figure 4 the same deformation tool is applied to two different objects: a straight ribbon and the twisted rods presented in Figure 2. The initial axis is a straight line while the final axis represents a node. A twist has also been added to this deformation.

A deformation tool can also be adapted to the object to which it will be applied. In Figure 5 we have applied the node-deformation tool to an object that is not straight, a horseshoe. This has been made possible simply by adapting the initial axis to the shape of the object. In Figure 5-left the visualized axis is the new initial one. The deformed horseshoe is shown in Figure 5-right together with the final axis.



Figure 4: The same deformation applied to two different objects.

4 Examples and Concluding Remarks

Two simple examples of deformations of well known objects by the AxDf technique are illustrated in Figure 6 and 7. Figure 6 presents two teapots. On top, an axis has been designed by the user inside the spout of the undeformed teapot. The deformed teapot presented below results from a deformation of the axis.

In Figure 7, a deformation that includes both a deformation of the axis and a modification of the scale graph is applied to the VW.

In our implementation, computing the deformation of the most detailed object (2160 vertices) takes approximatively 0.30 second on an IRIS 310 VGX.

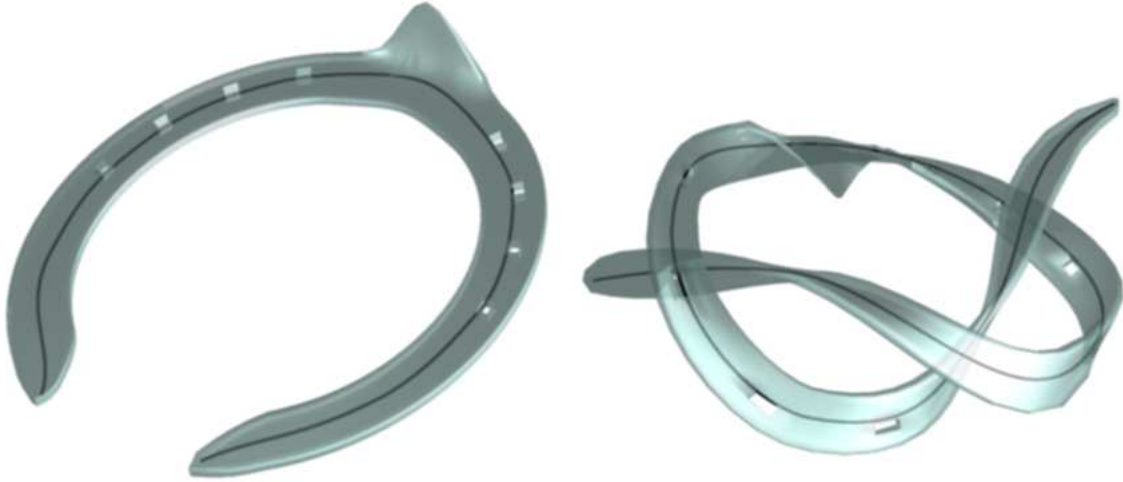


Figure 5: Adapting the deformation tool to the object.

Although specific and simple, the deformations that can be controlled with the AxDf technique are very common and most useful. Furthermore, AxDf offers the following advantages:

- Since the deformation is independent of the object to which it is applied, it can be re-used to deform other objects.
- Since AxDf is independent of the creation technique, it can be used to deform any existing object.
- AxDf can easily be combined with other deformation techniques, such as FFD.
- AxDf can be applied to many different geometric models such as spline surfaces, polygonal surfaces or hierarchical surfaces.

- AxDf can easily be integrated into most general and interactive modeling systems.
- AxDf is very intuitive and fully interactive.

This paper has shown that AxDf is a viable deformation technique for geometric modeling. It greatly increases the class of deformations obtained by manipulating a deformation entity and makes easier the control of some deformations usually defined using the FFD techniques. There are however a number of enhancements and extensions to AxDf that we should like to investigate. Some of them are:

- checking for self-intersection of the deformed surface,
- automatically (or semi-automatically) designing the first axis on the surface,
- implementing an adaptative subdivision technique such as that of Griessmair et al. [GP89] in order to maintain an acceptable resolution of the surface,
- allowing several axis to be used simultaneously for deforming an object.

Furthermore, due to the simplicity of the deformation entity (the axis), it seems likely that this approach would also be very attractive for animation applications.

AxDf is part of ACTION3D, a general interactive modeling system developed jointly by SOGITEC and INRIA.

References

- [Bar84] A.H. Barr. Global and Local Deformations of Solid Primitives. *SIGGRAPH'84, Computer Graphics*, 18(3):21–30, July 1984.
- [BB91] Paul Borrel and Dominique Bechmann. Deformation of n-Dimensional Objects. In *Symposium on Solid Modeling Foundations and CAD/CAM Applications*, pages 351–369. ACM, June 1991.

- [Cob84] B.S. Cobb. *Design of Sculptured Surfaces Using the B-Spline Representation*. PhD thesis, University of Utah, June 1984.
- [Coq90] Sabine Coquillart. EFFD: A Sculpturing Tool for 3d Geometric Modeling. *SIGGRAPH'90, Computer Graphics*, 24(4):187–196, August 1990.
- [FB88] D.R. Forsey and R.H. Bartels. Hierarchical B-Spline Refinement. *SIGGRAPH'88, Computer Graphics*, 22(4):205–212, August 1988.
- [GP89] J. Griessmair and W. Purgathofer. Deformation of Solids with Trivariate B-Splines. In *EUROGRAPHICS'89*, pages 137–148. North-Holland, 1989.
- [Klo86] F. Klok. Two Moving Coordinate Frames for Sweeping along a 3d Trajectory. *Computer Aided Geometric Design*, (3):217–229, 1986.
- [Par77] R.E. Parent. A System for Sculpting 3-D Data. *Computer Graphics*, 11(2):138–147, July 1977.
- [SP86] T.W. Sederberg and S.R. Parry. Free-Form Deformation of Solid Geometric Models. *SIGGRAPH'86, Computer Graphics*, 20(4):151–160, August 1986.
- [WBB⁺90] B. Wyvill, J. Bloomenthal, T. Beier, J. Blinn, A. Rockwood, and G. Wyvill. Modeling and Animating with Implicit Surfaces. *Siggraph Course Notes*, 23, 1990.
- [WW92] W. Welch and A. Witkin. Variational Surface Modeling. *SIGGRAPH'92, Computer Graphics*, 26(2):157–166, July 1992.



Figure 6: Deforming the teapot.



Figure 7: Deforming the VW.



Unité de recherche Inria Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 Villers Lès Nancy
Unité de recherche Inria Rennes, Irista, Campus universitaire de Beaulieu, 35042 Rennes Cedex
Unité de recherche Inria Rhône-Alpes, 46 avenue Félix Viallet, 38031 Grenoble Cedex 1
Unité de recherche Inria Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex
Unité de recherche Inria Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 Sophia-Antipolis Cedex

Éditeur

Inria, Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)

ISSN 0249-6399